Longword Access

Addition to PAGEMDMP

Fri, Mar 14, 2003

The Memory Dump page application has always allowed display of memory data by bytes or by 16-bit words. But the support for 32-bit longword access has not been included. This note discusses making that addition to the application code. The motivation is that there is now hardware that only responds to 32-bit accesses, and it would be useful if the generic memory dump page could be used to work with such hardware.

A recent addition to this application permits viewing floating point (32-bit single precision) data in memory, although it does not permit making settings of floating point values to memory. Perhaps this support can be expanded to include that for longword access, including writes.

There are only 32 columns on the little window console. Right now, the display format shows four 16-bit words per line. (The node number and address currently occupy 12 characters.) If we accessed the memory data by longwords, it would be helpful to illustrate that by displaying two longwords per line rather than four words per line. For example,

```
0          1          2          3
01234567890123456789012345678901

0509:401000 7F00 0001 0000 0000
09:00401000 7F00 0001 0000 0000

0509:00401000 7F000001 00000000
```

Note that by grouping the memory data into two longwords, we could display the full 32-bit memory address along with the full 16-bit node number. This feature might be a kind of bonus for adding the longword support. As it stands, we must decide between showing only a 2-digit node number with 8-digit address, or showing a 4-digit node number with 6-digit address.

But we must take care if this new format is used. Settings are currently permitted only for the cursor being in certain columns, namely those immediately following a 4-digit word value. We would need to expect only two possible columns for making longword writes. If pointers are to be followed, we must decide where the cursor can be to cause that; perhaps they should be in the same pair of positions as now, which are the middle two characters of the most significant word of the longword that is to be treated as a pointer value.

The means of selecting the longword format is easy; just extend the toggling of the BYTES, WORDS that follow the ACCESS BY text to include LONGS. In the Longword display format, one could not switch to 6-digit address; addresses would always be 8 digits long. The floating point option should still work, though, and perhaps we can also work out how to achieve floating point writes.

In the copy memory area of the display, access to longwords could also be done, merely by allowing the toggling of MB to MW to be extended to ML. This should be easy.

Implementation

A first implementation of this feature has been made in the IRM version of PAGEMDMP. It does not permit floating point settings, but it does permit longword settings in hexadecimal. One can also follow a pointer when in the longword access mode, but the pointer has to be within a longword.